



Instituto Tecnológico y de Estudios Superiores de Monterrey

**Programación de estructuras de datos y algoritmos  
fundamentales (TC1031.2)**

Profesor: Daniel Pérez Rojas

Act 6.2 - Reflexión Final de Actividades Integradoras de la  
Unidad de Formación TC1031 (Evidencia Competencia)

Daniela Hernández y Hernández A01730397

25 de octubre de 2020

Puebla, Pue.

## REFLEXIÓN FINAL (TC1031)

A lo largo de la Unidad de Formación TC1031 tuve que hacer uso de diversas estructuras de datos y algoritmos para completar las actividades integradoras del curso. Estas actividades estaban orientadas a dar solución a la Situación Problema, al desarrollar código eficiente y escalable. Para ello, tuve que considerar métodos de optimización del tiempo de ejecución y recursos computacionales, los cuales involucran utilizar estructuras y algoritmos fundamentales.

De manera resumida, a través de las cinco actividades integradoras se utilizó un archivo de texto que contenía una gran cantidad de registros de acceso a IPs, junto con la fecha y mensaje de éstos. Este archivo debía procesarse, ordenarse y estructurarse de maneras específicas en cada actividad. Empezando por la 1.3, en esta actividad se hizo uso de mergeSort, el cual es uno de los algoritmos de ordenamiento más eficientes, teniendo una complejidad de tiempo de  $O(n \log n)$ . Además, se utilizó una búsqueda lineal para imprimir las IPs dentro de un rango determinado, la cual a diferencia del primer algoritmo no es muy eficiente, teniendo una complejidad lineal  $O(n)$ .

La siguiente actividad, la 2.3, retomó la idea de la actividad previa, pero en esta ocasión los registros se iban insertando de manera ordenada en una lista doblemente ligada. No obstante, este acercamiento no mejoró mucho el tiempo de ejecución, puesto que las operaciones de inserción y búsqueda siguen teniendo complejidad  $O(n)$ . Por tal motivo, en la actividad 3.4 se implementó un árbol de búsqueda binario, cuyas operaciones reducen esta complejidad a  $O(\log n)$ . Esto representa una gran mejora en la eficiencia del algoritmo y nos acerca más a una solución óptima del problema.

Después de ésta, en la actividad 4.3 se utilizó un grafo para reestructurar las IPs y sus accesos de manera menos repetitiva y más jerárquica, a la vez que permitía contabilizar los hosts y registros de cada red. Sin embargo, el tiempo de ejecución de las operaciones regresó a  $O(n)$ , ya que al usar listas de adyacencias, éstas se deben recorrer para insertar o encontrar información. No obstante, en la actividad integradora final, la 5.2 se hizo una mejora enorme al incluir funciones y tablas hash, ya que esto permite hacer la inserción y la búsqueda en tiempo constante  $O(1)$ , siendo esta última la opción más eficiente para resolver la situación problema. Aún así, este proceso se podría mejorar aún más, ya que el algoritmo para construir

el grafo y convertirlo posteriormente a una tabla hash sigue consumiendo una cantidad considerable de recursos.

Una posible solución para mejorar estas opciones, si lo que se busca es detectar aquellas redes con mayor número de incidentes, podría ser retomar la idea de lo que se hizo en la actividad 4.3 con el grafo, pero esta vez implementarlo como un *max heap*, donde el parámetro de ordenamiento sería el número de accesos encontrados para cada red. Como resultado, la búsqueda de las redes sería muy eficiente, teniendo una complejidad constante  $O(1)$  y operación como inserción y eliminación con complejidades logarítmicas  $O(\log n)$  o lineales  $O(n)$ , en el peor caso.

Ahora, hablando sobre las subcompetencias de esta materia y mi desempeño en ellas, considero que logré alcanzar el nivel esperado. En la subcompetencia SICT0301 B fui capaz de identificar aquellas partes de mi código que generarían un aumento en el tiempo de ejecución lo que me permitió reconocer y buscar soluciones más optimizadas y que minimizarían el uso de recursos del sistema computacional.

Por otra parte, en la SICT0302 B, demostré entendimiento sobre conceptos abstractos como las diversas estructuras de datos, y logré convertir estos conocimientos en algo aplicable y práctico. Además, utilicé plataformas informativas confiables para profundizar en temas como la complejidad de tiempo, de espacio y notación BigO de mi implementación, a la vez que empecé a usar herramientas tecnológicas como GitHub, para tener mejor control de mi código.

En cuanto a la última competencia, la SICT0303 B, pienso que ocupé al seleccionar el algoritmo o estructura de datos adecuado para cubrir las necesidades de los problemas que se nos presentaron. Igualmente, durante todo el curso me preocupé por escribir código significativo, completo y óptimo, lo cual brindaría facilidad para entenderlo por otras personas o por mí misma en un futuro.

En conclusión, considero que logré desarrollar las subcompetencias asociadas a esta unidad de información al implementar soluciones eficientes y fundamentadas en los conceptos más importantes de las ciencias computacionales como lo son las estructuras de datos y los algoritmos. Además, pienso que obtuve una base sólida para continuar aprendiendo conceptos de programación más complejos y aplicar estos conocimientos en otros problemas o situaciones de mi vida académica y profesional.